

Installing OpenBiblio

Contents

1	System requirements	1
2	Installing OpenBiblio 1.0 on a Windows computer	1
2.1	Install prerequisite software	1
2.2	Install OpenBiblio	2
2.3	Using the OpenBiblio installer.	2
3	Installing OpenBiblio 1.0 on a shared Linux host	2
4	Installing OpenBiblio 1.0 on a Linux host with full access	2
4.1	Install prerequisite software	2
4.2	Install yaz (optional)	3
4.3	Start the Apache and MySQL services	3
4.4	Create a MySQL database	3
4.5	Prepare OpenBiblio for its installation	3
4.6	Using the OpenBiblio installer.	4

System requirements

We have tested OpenBiblio installation on Windows 7 and Ubuntu Linux 14.04.

We also plan to test other versions of Windows, Ubuntu Linux 12.04, Debian Linux, Fedora Linux, Mac OSX, and CentOS Linux.

To install OpenBiblio, you will need PHP version 5.4 or higher. We are actively revising the codebase to make OpenBiblio fully compatible with PHP 7. You are welcome to use OpenBiblio with PHP 7, but be aware that you may see warning messages.

You will need MySQL. Not sure of version.

You will also need some Web server program. OpenBiblio currently only works with Apache or with FastCGI implementations on other Web server software.

Installing OpenBiblio 1.0 on a Windows computer

You can turn your personal Windows computer into an OpenBiblio server. You will need administrator access to your computer, an Internet connection, and you may also need to check with your local IT experts that you are not behind a restrictive firewall.

Note that turning your personal computer into a server carries some risks, especially if your server is broadcasting across the web instead of just a local network. We recommend that you run this software on a dedicated computer that does not contain any sensitive or private data.

Install prerequisite software

1. Download EasyPHP's Webserver from <http://easyphp.org>. We have tested these instructions with version 14.1 of EasyPHP.
 2. Agree to the relevant licenses and choose an installation folder that is convenient for you.
 3. Once EasyPHP is installed, it will appear as an icon in your system tray. On Windows 7, click on this icon and open the dashboard.
 4. The dashboard will open in your default Web browser. Once it does, click Settings in the top right corner of the screen.
 5. Click on HTTP SERVER. Install and start this service.
 6. Click on DB SERVER. Install and start this service.
 7. Go back to the dashboard.
 8. Under modules, you should now see an option to administer MySQL through a program called PhpMyAdmin.
 9. Open PhpMyAdmin. The default username is root and there is no default password. Click on Users.
 10. Click on "Edit Privileges" next to root.
 11. Click on "Change Password".
 12. Click on Add user.
 13. Choose a memorable username, like *obiblio*, and enter it. We suggest using a name that only contains letters, numbers, and underscore characters.
 14. In the host field, enter "localhost".
 15. Create a long, secure, memorable password.
 16. Under "Database for user", there should be a checkbox for "Create database with same name and grant all privileges". Check this box.
 17. Click go.
-

Install OpenBiblio

1. Open the EasyPHP dashboard.
2. Go to Settings and open the HTTP SERVER settings.
3. Note the Document Root field. This is the folder on your computer where OpenBiblio will be running.
4. Open the DB SERVER settings. Note the parameters area.
5. Download OpenBiblio and place the files into the Document Root folder.
6. Go back to the EasyPHP dashboard and open the HTTP SERVER settings once more.
7. There is a link there to open up OpenBiblio under the URL field. Click on it. The installer should appear.

Using the OpenBiblio installer.

1. Fill out the required information using the parameters of your MySQL setup. The hostname should be the IP address and port in the format 192.168.333.333:3388, where 192.168.333.333 is the IP address and 3388 is port number.
2. Install test data if you would like.
3. Log in to your new OpenBiblio system. The default user is admin with password admin.

Installing OpenBiblio 1.0 on a shared Linux host

These instructions will help you set up OpenBiblio on a Web server provided by Dreamhost, Hostgator, or a similar service.

1. Create a database using your hosting service's interface. Note the host, username, password, and database name.
2. Download the Filezilla FTP program

Installing OpenBiblio 1.0 on a Linux host with full access

These instructions consist mainly of commands that you can run in your terminal to quickly get a working installation of OpenBiblio. If you are not comfortable with using the terminal, feel free to use the instructions above for a shared Linux host.

These procedures have been tested on Fedora 21 and Ubuntu 14.04.

Install prerequisite software

OpenBiblio requires apache, PHP, and MySQL. If you intend to copy catalog records from other libraries, we strongly recommend that you use yaz. The following steps include a YAZ install.

On Ubuntu 14.04:

```
sudo apt-get install apache2 php5 mysql-server yaz php5-dev php5-mysql php-pear libyaz4-dev <↵  
unzip
```

Tip

This will prompt you to choose a root password for MySQL. Make sure it is a secure one!

On Fedora 21:

```
sudo yum install httpd php mysql-server php-devel php-mysql php-pear unzip yaz libyaz-devel ←  
gcc  
sudo mysql_secure_installation
```

Tip

mysql_secure_installation will ask you for a current root password, which you may leave blank. Be sure to give root a new, secure password. You should also choose to remove anonymous users, disallow root login remotely, and remove the test database.

Install yaz (optional)

Next, install yaz as a PHP extension using pecl.

```
sudo pecl install yaz
```

Add a new line to your php.ini file. This file can be found at /etc/php/php.ini in Fedora and /etc/php5/apache/php.ini in Ubuntu. The line should be something like the following:

```
extension=yaz.so
```

Start the Apache and MySQL services

On Fedora 21:

```
sudo service httpd start  
sudo service mysqld start
```

On Ubuntu:

```
sudo service apache2 start  
sudo service mysql start
```

Create a MySQL database

First, open up MySQL.

```
mysql -uroot -p #MySQL will ask you for the root password you just created
```

Next, type the following commands to create a user and database for OpenBiblio to use.

```
CREATE DATABASE obiblio;  
GRANT ALL PRIVILEGES ON obiblio.* TO obib_user@localhost IDENTIFIED BY 'obib_password';
```

Prepare OpenBiblio for its installation

```
wget -O obib.zip https://bitbucket.org/mstetson/obiblio-1.0-wip/get/26f88507b580.zip  
unzip obib.zip
```

Delete the default "It Works" page from /var/www/html.

```
sudo mv ./* /var/www/html/
```

On Fedora 21, you may need to run the following command to configure SELinux to allow access to the folder.

```
restorecon -r /var/www/html
```

Using the OpenBiblio installer.

1. Fill out the required information using the parameters of your MySQL setup. The hostname should be the IP address and port in the format 192.168.333.333:3388, where 192.168.333.333 is the IP address and 3388 is port number.
2. Install test data if you would like.
3. Log in to your new OpenBiblio system. The default user is admin with password admin.

To protect sensitive member data, we strongly encourage you to set up HTTPS, so that anybody who intercepts OpenBiblio data will see an encrypted jumble, rather than your members' information.

The following is extracted from "Ubuntu Unleashed", 2016 edition, Sams, pp 513 - 515.

All below assumes you are using Apache2 running on a Ubuntu server.

1. Enable the following module: `sudo a2enmod ssl`

```
to configure Apache2 for HTTPS using the default configuration for testing:
sudo a2ensite default-ssl
and restart using
sudo service apache2 restart
```

2. create a self-signed certificate and key first generate a key `openssl genrsa -des3 -out server.key 2048`

```
next a signing request (CSR)
openssl req -new -key server.key -out server-csr
```

```
generate a certificate for 1 year (365 days) larger numbers are allowed - I ↔
use 36500!
openssl x509 -req -days 365 -in server.csr -signkey server.key -out server.crt
```

```
copy to the certificate to a 'proper' location
sudo cp server.crt /etc/ssl/certs/
```

```
copy the key to a proper location
sudo cp server.key /etc/ssl/private/
```

```
edit apache2's default -ssl file /etc/apache2/sites-available/default-ssl to ↔
read as
SSLEngine on
SSLCertificateFile /etc/ssl/certs/server.crt
SSLCertificateKeyFile /etc/ssl/private/server.key
```

```
to configure Apache2 to use this info
sudo a2ensite default-ssl
```

Now restart Apache; be prepared to input the certificate's key password.
Be sure to save this password/phrase, you will need it every time you restart ↔
Apache.
`sudo service apache2 restart`